

# ANALISIS SENTIMEN ULASAN PENGUNJUNG SIRKUIT MANDALIKA PADA *GOOGLE MAPS* DENGAN METODE NAÏVE BAYES

Salsabila Mariska<sup>1</sup>, I Made Dedy Setiawan<sup>2</sup>

<sup>1,2</sup>Institut Bisnis dan Teknologi Indonesia

salsabila.mariska02@gmail.com \*, <sup>2</sup>dedy.setiawan@instiki.ac.id

Corresponding Author: Salsabila Mariska

## ABSTRACT

Mandalika Circuit is one of the leading sports tourism destinations in Indonesia that attracts many reviews from visitors through the Google Maps platform. This research aims to analyze the sentiment of visitor reviews of the Mandalika Circuit using the Naive Bayes method. Review data is obtained through an automatic crawling process from Google Maps, then a text preprocessing stage is carried out which includes cleaning, case folding, stopword removal, normalization, tokenization, and lemmatization. Sentiment labeling is done using a dictionary-based lexicon approach of positive and negative words, and word weighting using the TF-IDF method. The classification model is built using the Multinomial Naive Bayes algorithm with 80% data division as training data and 20% as test data. The model evaluation results showed good performance with an accuracy of 70.96%, precision of 74.66%, recall of 70.96%, and F1-score of 69.84%. Word Cloud visualization shows that positive reviews dominate, followed by negative and neutral sentiments. These results provide insight for Mandalika Circuit managers in improving services and visitor experience. This research can also be used as a reference for the development of similar research in the tourism sector based on digital reviews.

**Keywords:** *Google Maps, Mandalika, Naive Bayes, Sentiment Analysis*

## ABSTRAK

Sirkuit Mandalika merupakan salah satu destinasi sports tourism unggulan di Indonesia yang menarik banyak ulasan dari pengunjung melalui platform Google Maps. Penelitian ini bertujuan untuk menganalisis sentimen ulasan pengunjung terhadap Sirkuit Mandalika menggunakan metode Naive Bayes. Data ulasan diperoleh melalui proses crawling otomatis dari Google Maps, kemudian dilakukan tahap preprocessing teks yang meliputi cleaning, case folding, stopword removal, normalisasi, tokenisasi, dan lemmatization. Labeling sentimen dilakukan menggunakan pendekatan lexicon berbasis kamus kata positif dan negatif, serta pembobotan kata menggunakan metode TF-IDF. Model klasifikasi dibangun menggunakan algoritma Multinomial Naive Bayes dengan pembagian data 80% sebagai data latih dan 20% sebagai data uji. Hasil evaluasi model menunjukkan performa yang cukup baik dengan akurasi sebesar 70,96%, precision sebesar 74,66%, recall sebesar 70,96%, dan F1-score sebesar 69,84%. Visualisasi Word Cloud menunjukkan bahwa ulasan positif mendominasi, diikuti oleh sentimen negatif dan netral. Hasil ini memberikan wawasan bagi pengelola Sirkuit Mandalika dalam meningkatkan layanan dan pengalaman pengunjung. Penelitian ini juga dapat dijadikan referensi untuk pengembangan riset serupa di sektor pariwisata berbasis ulasan digital.

**Kata Kunci:** *Google Maps, Mandalika, Naive Bayes, Analisis Sentimen*

## 1. Pendahuluan

Sirkuit Mandalika terletak di Lombok, Nusa Tenggara Barat, dan merupakan salah satu proyek unggulan pemerintah Indonesia dalam mempromosikan *sports tourism*. Resmi dibuka pada tahun 2021, sirkuit ini menjadi salah satu destinasi utama untuk ajang balap internasional seperti MotoGP dan *World Superbike* (WSBK). Dengan panjang lintasan 4,3 kilometer dan 17 tikungan, Sirkuit Mandalika menawarkan pemandangan yang spektakuler karena terletak di dekat pantai dengan latar pegunungan yang indah. Sejak penyelenggaraan ajang balap MotoGP



dan kejuaraan dunia lainnya, sirkuit ini tidak hanya menjadi simbol kemajuan infrastruktur olahraga di Indonesia tetapi juga menarik perhatian wisatawan domestik dan mancanegara. Sebagai bagian dari pengembangan Kawasan Ekonomi Khusus (KEK) Mandalika, keberadaan sirkuit ini diharapkan mampu meningkatkan perekonomian lokal dan memperkuat citra Indonesia di kancah internasional (Europa Publications, 2022).

*Google Maps* merupakan *platform* yang sering digunakan oleh pengunjung untuk memberi ulasan mengenai pengalaman mereka. Haq (2020) menyatakan Fitur pemberian *review* pada *Google Maps* merupakan salah satu hal dari era *big data* yang pada saat ini setiap orang dapat menaruh jejak setelah mereka mendatangi sebuah tempat (Irvandi dkk., 2023). *Google Maps* memungkinkan pengguna untuk mencari dan menampilkan berbagai informasi tentang suatu lokasi. Salah satu informasi yang tersedia adalah ulasan dari para pengunjung, yang dapat menjadi referensi bagi calon pengunjung lainnya. Menariknya, sebanyak 78% orang lebih mempercayai ulasan *online* dibandingkan dengan rekomendasi pribadi dari seseorang (Ipmawati dkk., 2024). *Google Maps* memberi peluang bagi pengunjung untuk lebih berpartisipasi dengan memberikan ulasan sertalangsung mengenai pengalaman mereka di tempat tersebut. Pengunjung dapat dengan mudah memberikan umpan balik positif atau kritik konstruktif melalui platform ini, yang dapat membantu pengelola tempat wisata untuk terus meningkatkan kualitas pengalaman yang mereka tawarkan (Prakoso & Hermawan, 2023). Ulasan ini mencerminkan berbagai pengalaman dan pendapat dari pengunjung mengenai pelayanan, kebersihan, fasilitas sirkuit dan juga kepuasan pengunjung. Serta ulasan yang diberikan sering kali tidak sesuai dengan rating yang tercantum. Ulasan-ulasannya mengandung sentimen yang dapat digunakan untuk mengukur tingkat kepuasan dan mengidentifikasi area yang perlu diperbaiki. Untuk memahami sentimen di balik ulasan-ulasan ini, diperlukan analisis lebih lanjut salah satunya dengan melakukan analisis sentimen.

Analisis sentimen adalah langkah penting dalam mengungkapkan apakah ulasan tersebut bersifat positif atau negatif. Penggunaan berbagai model *Machine Learning* akan memungkinkan kita untuk mengidentifikasi pola-pola dalam ulasan, termasuk kata-kata kunci yang menggambarkan pengalaman positif atau negatif (Prakoso dan Hermawan, 2023). Dengan mengaplikasikan metode analisis sentimen pada ulasan *Google Maps*, pihak terkait dapat memperoleh informasi yang lebih objektif dan berbasis data dalam merencanakan kebijakan atau perbaikan layanan. Jumlah ulasan yang besar dan beragam membuat analisis manual menjadi sulit dan tidak efisien. Oleh karena itu, diperlukan metode otomatis untuk menganalisis sentimen dari ulasan-ulasan tersebut agar pengelola dapat memahami dengan lebih mudah apakah ulasan cenderung positif atau negatif. Salah satu metode yang efektif untuk analisis sentimen adalah metode *Naive Bayes*, yang mengklasifikasikan teks berdasarkan probabilitas kemunculan kata-kata tertentu yang berhubungan dengan kategori sentimen.

*Naive Bayes* adalah metode yang sering digunakan dalam analisis teks karena kesederhanaannya, kecepatan pemrosesan, dan akurasi yang cukup baik, terutama dalam mengolah data yang besar. Metode *Naive Bayes* ini banyak digunakan karena metode ini dapat menghasilkan akurasi yang baik dan stabil (Rizkina dan Hasan, 2023). Dengan metode ini, sistem dapat diatur untuk mengidentifikasi sentimen dalam ulasan berdasarkan pola kata-kata tertentu, yang kemudian dikategorikan ke dalam sentimen positif atau negatif. Hasil analisis sentimen ini diharapkan dapat memberikan wawasan mendalam bagi pengelola Sirkuit Mandalika mengenai aspek-aspek yang disukai dan yang perlu ditingkatkan, sehingga dapat meningkatkan kepuasan pengunjung di masa depan.

Penelitian ini bertujuan untuk menerapkan metode *Naive Bayes* dalam menganalisis sentimen ulasan pengunjung Sirkuit Mandalika di *Google Maps*, guna memahami persepsi umum dan tanggapan pengunjung terhadap fasilitas dan pengalaman yang diberikan oleh Sirkuit Mandalika ini.



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

## 2. Bahan & Metode

Penulis melakukan penelitian di media informasi *Google Maps* dan melakukan pengambilan data dari tahun 2021-2025, jangka waktu tersebut digunakan karena pada tahun tersebut Sirkuit Mandalika mulai beroperasi 12 November 2021. Dalam analisis sentimen ulasan pengunjung sirkuit mandalika pada penelitian ini penulis melaksanakan proses pengumpulan data sebagai berikut

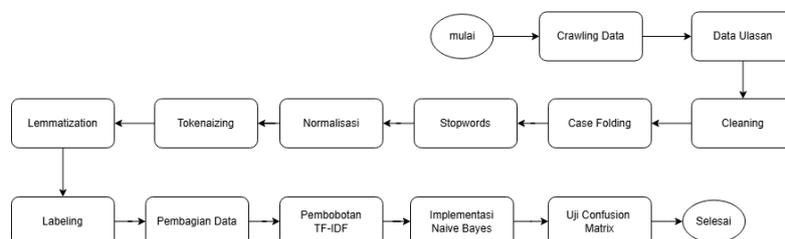
Pengambilan data primer dilakukan dengan *crawling* data dalam pengambilan data ulasan tersebut penulis melakukan pengambilan data menggunakan *Instant Data Scraper*, sebuah ekstensi browser yang memungkinkan pengambilan data secara otomatis dari halaman *web*. Data ulasan pada *Google Maps* diambil dengan memanfaatkan alat ini, yang secara otomatis mengekstrak informasi seperti nama pengguna, isi ulasan, tanggal, dan rating. Proses ini dilakukan secara sistematis untuk memastikan data yang relevan dengan penelitian dapat diperoleh dengan cepat dan efisien. Berikut merupakan gambar *instant data scraper* pada gambar 1.



Gambar 1 Logo *Instant Data Scraper*

Pengumpulan data sekunder pada penelitian ini menggunakan observasi non-partisipasi. Observasi non-partisipasi merupakan metode pengumpulan data yang dilakukan dengan mengamati tanpa ikut serta secara langsung dalam aktivitas yang sedang diamati. Peneliti memperoleh data penelitian dari sumber data sekunder secara tidak langsung melalui media perantara (Rohmania dan Abidin, 2024). Data sekunder yang digunakan dalam penelitian ini berasal dari berbagai sumber seperti jurnal, artikel, skripsi, buku, dan sumber *online* yang dinilai relevan dan terpercaya. Penggunaan data ini diharapkan dapat memperkuat argumen penelitian dan memberikan dasar yang lebih komprehensif untuk analisis.

Proses penelitian ini akan dilakukan dengan mengikuti tahapan yang telah dirancang, sebagaimana dijelaskan pada Gambar 2 di bawah ini:



Gambar 2 Tahapan Penelitian

Penerapan teknik *web crawling* menggunakan alat *Instant Data Scraper*. Alat ini dipilih karena kemampuannya secara otomatis mengambil data dari berbagai sumber yang sesuai dengan kebutuhan penelitian. Dengan bantuan *Instant Data Scraper*, proses ekstraksi data menjadi lebih efisien dan terorganisir, sehingga mendukung analisis data secara lebih efektif.



Tahap *Cleanning* merupakan langkah untuk menghapus kata-kata yang tidak diperlukan dari teks, dengan tujuan memaksimalkan dan mengurangi munculnya *noise* pada tahap klasifikasi. *Cleanning* juga dapat diartikan sebagai proses menghapus *noise* yang berupa *emoticon* dan karakter yang tidak signifikan dalam kalimat ulasan. *Case folding* adalah proses mengubah semua huruf dalam teks menjadi huruf kecil untuk normalisasi data. Proses ini dilakukan untuk menghilangkan perbedaan antara huruf kapital dan huruf kecil, sehingga analisis teks dapat dilakukan dengan lebih konsisten, terutama pada data teks seperti ulasan, artikel, atau dokumen lainnya.

*Stopword removal* adalah proses menghapus kata-kata umum yang tidak memiliki makna penting dalam analisis teks. Stopwords adalah kata-kata seperti “dan”, “yang”, “di”, atau “untuk”, yang sering muncul dalam teks tetapi biasanya tidak menambah nilai dalam analisis. *Normalisasi* adalah salah satu langkah penting dalam prapemrosesan teks, yang bertujuan untuk menyetarakan teks ke dalam bentuk standar agar lebih konsisten dan mudah diproses.

*Tokenizing* adalah proses membagi teks menjadi unit-unit kecil yang disebut token. Token dapat berupa kata, frasa, karakter, atau simbol. Strategi dasarnya melibatkan identifikasi penanda antara kalimat, seperti tanda titik (.), karakter baris baru (/n), dan terkadang tanda titik koma (;). *Lemmatization* adalah proses dalam pengolahan bahasa alami (*Natural Language Processing* atau NLP) yang digunakan untuk mengubah kata-kata ke bentuk dasarnya atau lemma-nya.

Pada tahap pelabelan, menggunakan metode *InSet Lexicon*, di mana daftar kata atau frasa dari kamus sentimen digunakan untuk secara otomatis memberikan label pada teks berdasarkan polaritasnya. Tahap ini memuat kamus yang berisi kata-kata yang sudah memiliki nilai sentimen yang akan digunakan pada saat tahap ekstraksi kata kunci dan penentuan sentimen. Kamus yang digunakan adalah kamus kata kunci positif dan kamus kata kunci negatif berbahasa Indonesia (Undap dkk., t.t.). Proses pelabelan dimulai dengan mempersiapkan teks melalui tahap *preprocessing*, seperti *tokenisasi*, normalisasi, dan pembersihan data. Selanjutnya, setiap kata dalam teks diperiksa terhadap daftar kata dalam *lexicon*, dan jika ditemukan kecocokan, kata tersebut akan diberi label sesuai kategori atau bobot yang terkait. Berikut beberapa contoh kamus *InSet Lexicon* pada tabel 1.

Tabel 1 Kamus *InSet Lexicon*

No	Kata	<i>Polarity score</i>	Kata	<i>Polarity Score</i>
1	selenggara	3	kalah	-4
2	pesona	4	debu	-3
3	Citra	2	doang	-1
4	Jatuh	1	bodo	-5
5	Unggulan	5	Hujan	-2

TF-IDF (*Term Frequency-Inverse Document Frequency*) adalah metode pembobotan yang digunakan untuk mengukur pentingnya suatu kata dalam sebuah dokumen dibandingkan dengan keseluruhan kumpulan dokumen (*corpus*). Metode ini menggabungkan dua komponen utama, yaitu *Term Frequency* (TF), yang menghitung frekuensi kemunculan sebuah kata dalam dokumen tertentu, dan *Inverse Document Frequency* (IDF), yang mengukur seberapa jarang kata tersebut muncul di seluruh dokumen dalam corpus. Kata yang sering muncul di dokumen tertentu tetapi jarang ditemukan di dokumen lain akan memiliki



bobot TF-IDF yang tinggi, sementara kata-kata umum yang sering muncul di banyak dokumen, seperti *stop words*, akan memiliki bobot rendah. Hasil dari TF-IDF adalah refrenansi numerik kata-kata dalam dokumen, yang akan digunakan untuk tahap analisis sentimen atau pemodelan data lebih lanjut.

Langkah selanjutnya yaitu membagi data latih dan data uji. Data dibagi menggunakan perbandingan 80:20, dengan data latih 80% dari setiap kategori dan data uji 20% dari setiap kategori. Data latih digunakan untuk membentuk sebuah model *classifier*. Semakin besar data latih yang digunakan, maka semakin baik machine dalam memahami pola data. Dan data uji digunakan untuk mengukur sejauh mana *classifier* berhasil melakukan klasifikasi dengan benar.

*Naive Bayes* memiliki keunggulan dalam kecepatan dan efisiensi komputasi, yang membuatnya kandidat yang ideal untuk aplikasi yang membutuhkan dataset yang besar (Haq et al., 2024). Algoritma ini berlandaskan *teorema Bayes*, yang menghitung probabilitas sebuah kelas berdasarkan informasi yang tersedia, dengan asumsi bahwa setiap fitur saling independen. Salah satu tugas *Data Mining* adalah klasifikasi data, yaitu memetakan (mengklasifikasikan) data ke dalam satu kelas atau beberapa kelas yang sudah di definisikan sebelumnya. Salah satu metoda dalam klasifikasi data adalah *Naive Bayes* yaitu salah satu metode *machine learning* yang memanfaatkan perhitungan probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, cara kerja *Naive Bayes* yaitu memprediksi probabilitas di masa depan berdasarkan pengalaman di masa sebelumnya (Darwis dkk., 2021).

Pada tahap evaluasi model atau pengujian, akurasi (*accuracy*), presisi (*precision*), *recall*, dan *F1-Score* dihitung menggunakan *confusion matrix* pada ulasan yang telah diklasifikasikan dengan algoritma *Naive Bayes*. Evaluasi ini bertujuan untuk menentukan sejauh mana model yang dibangun mampu secara akurat mengklasifikasikan dokumen menjadi sentimen positif dan negatif. Dengan demikian, proses evaluasi ini memberikan gambaran menyeluruh tentang kinerja dan efektivitas model dalam mengenali serta membedakan berbagai jenis sentimen dalam ulasan (Fatikhah dkk., 2024).

### 3. Hasil

Pengumpulan data dalam penelitian ini dilakukan dengan memanfaatkan *Instant Data Scraper*, yaitu sebuah ekstensi pada peramban *Google Chrome* yang memungkinkan pengambilan data secara otomatis dari laman web. Peneliti mengakses halaman *Google Maps* yang memuat ulasan-ulasan pengunjung terhadap Sirkuit Mandalika, kemudian menjalankan *Instant Data Scraper* untuk mengekstraksi informasi penting seperti nama pengguna, tanggal ulasan, serta isi komentar. Seluruh data yang berhasil diperoleh selanjutnya diekspor dalam format *.csv* dan digunakan sebagai dasar dalam proses analisis data pada tahap berikutnya.

*Preprocessing* data merupakan proses awal yang bertujuan untuk membersihkan dan mempersiapkan data mentah agar lebih mudah dan akurat untuk dianalisis. *Preprocessing* meliputi menghilangkan kata yang tidak diperlukan seperti simbol, karakter atau emoji, tanda baca dan mengubah huruf besar menjadi huruf kecil. Proses pembersihan ini akan digunakan nantinya untuk melakukan pengujian yang dimana akan mengevaluasi hasil dari ulasan pengunjung terhadap Sirkuit Mandalika. Langkah – langkah *preprocessing* meliputi beberapa tahapan seperti *cleaning*, *case folding*, *stopword*, *normalisas*, *tokenaizing* dan *lemmatization*.

Proses *cleaning* dilakukan untuk membersihkan data dari elemen-elemen yang tidak relevan atau mengganggu analisis. Proses ini mencakup penghapusan karakter khusus, tanda baca, angka, emotikon, serta data duplikat yang terdapat dalam teks ulasan. Proses ini



bertujuan untuk menghasilkan data yang bersih dan konsisten agar analisis sentimen dapat dilakukan secara optimal.

Tahap selanjutnya yaitu *case folding* yang dilakukan untuk menyeragamkan semua teks ulasan dengan cara mengubah seluruh huruf menjadi huruf kecil (*lowercase*). Berikut merupakan proses *case folding* dapat dilihat pada gambar 3 berikut.

```
#case folding
def casefolding(text):
    # Check if text is a string before applying lower()
    if isinstance(text, str):
        text = text.lower()
    # If not a string, return it as is to avoid the error
    return text
```

Gambar 3 Proses *Case Folding*

Setelah dilakukan *case folding*, tahap selanjutnya yaitu, *stopword* proses menghapus kata-kata umum yang tidak memiliki makna penting dalam teks. Proses *normalisasi* dilakukan untuk mengubah kata-kata tidak baku, singkatan, serta bentuk informal lainnya menjadi bentuk baku sesuai dengan kaidah bahasa Indonesia. Contohnya, kata "gk" atau "ga" akan dinormalisasi menjadi "tidak", dan "bgt" menjadi "banget". Tahapan ini penting untuk memastikan konsistensi data dan meningkatkan kualitas analisis, khususnya dalam pemrosesan teks yang berbasis pada pencocokan kata atau pembobotan kata kunci.

Tahap *tokenaizing* yaitu proses membagi teks menjadi unit-unit kecil. Token dapat berupa kata, frasa, karakter, atau simbol. Berikut merupakan proses *tokenaizing* dapat dilihat pada gambar 4 berikut.

```
def tokenization(text):
    if isinstance(text, str):
        tokens = text.split()
        return tokens

# Changed 'stopword' to 'stopwords' to match the actual column name
df['Tokenization'] = df['stopwords'].apply(tokenization)
df.head(10)
```

Gambar 4 Proses *Tokenaizing*

Tahapan terakhir dalam *preprocessing* ini adalah *lemmatization* proses dalam pengolahan bahasa alami (Natural Language Processing atau NLP) yang digunakan untuk mengubah kata-kata ke bentuk dasarnya. Setelah tahapan *preprocessing* selesai, selanjutnya yaitu proses pelabelan data yang dilakukan secara semi-otomatis dengan pendekatan berbasis *lexicon*, yaitu menggunakan kamus kata positif dan kamus kata negatif yang telah disusun sebelumnya. Kamus ini berisi daftar kata-kata yang secara umum mengandung makna positif maupun negatif beserta bobot nilai sentimennya masing-masing. Setiap kata dalam data akan dibandingkan dengan kata-kata dalam kamus, jika ditemukan kecocokan, maka bobotnya akan dijumlahkan atau dikurangkan untuk menghasilkan nilai akhir yang disebut *polarity score*. Berdasarkan nilai ini, sentimen dari suatu teks ditentukan, jika skor lebih dari nol maka dikategorikan sebagai *positive*, jika kurang dari nol sebagai *negative*, dan jika sama dengan nol sebagai *neutral*.

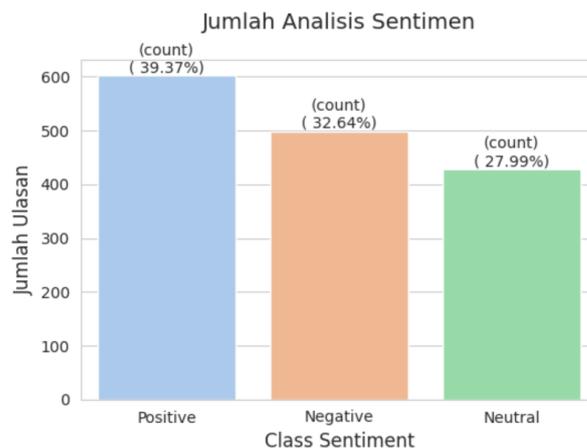


Setelah tahapan pelabelan, kita dapat mengetahui berapa jumlah sentimen positif, negatif, dan netral yang dapat kita lihat pada gambar 5.

Sentiment	
Positive	602
Negative	499
Neutral	428

Gambar 5 Hasil Sentimen

Dari hasil pelabelan menggunakan pendekatan *inSet lexicon*, diketahui bahwa hasil sentimen positif sebanyak 602, negatif sebanyak 499, dan netral sebanyak 428 data. Berikut merupakan hasil sentimen dalam bentuk diagram batang dapat dilihat pada gambar 6 di bawah ini:



Gambar 6 Hasil Presentase Sentimen Positif, Negatif, dan Netral Dalam Bentuk Diagram Batang

Setelah proses *preprocessing* dan pelabelan dilakukan, tahap selanjutnya adalah pembobotan kata menggunakan metode TF-IDF (*Term Frequency–Inverse Document Frequency*). TF-IDF digunakan untuk menghitung tingkat kepentingan sebuah kata dalam suatu dokumen relatif terhadap seluruh kumpulan dokumen. Semakin sering sebuah kata muncul dalam satu dokumen, tetapi jarang muncul di dokumen lain, maka nilainya akan semakin tinggi, tetapi jika kata tersebut muncul di banyak dokumen, maka bobotnya akan berkurang. Berikut merupakan proses pembobotan TF-IDF yang ditunjukkan pada gambar 7 di bawah ini:

```
# proses TF-IDF
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer

# Inisialisasi TfidfVectorizer
vectorizer_tfidf = TfidfVectorizer() # create an instance of TfidfVectorizer

# Transform Data untuk TF-IDF
word_tfidf = vectorizer_tfidf.fit_transform(data["Tokenization"].values)
word_tfidf
```

Gambar 7 Proses Pembobotan TF-IDF

Kode ini menggunakan TF-IDF dengan memanfaatkan fungsi *TfidfVectorizer()* dari pustaka *scikit-learn*. Proses ini dilakukan terhadap data yang telah melalui tahap tokenisasi, yang terdapat pada kolom *Tokenization*. Selanjutnya, metode *fit\_transform()* diterapkan untuk mengubah kumpulan teks tersebut menjadi matriks TF-IDF, di mana setiap nilai dalam matriks



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

merepresentasikan bobot dari suatu kata berdasarkan frekuensinya di dalam dokumen serta keseluruhan korpus. Hasil pembobotan ini digunakan sebagai masukan untuk tahap klasifikasi sentimen.

Data akan dibagi menjadi 2 yaitu data latih dan data uji. Dalam penelitian ini penulis akan membuat pembagian data dengan rasio 80:20, yang dimana 80% data latih dan 20% data uji.

Berikut merupakan tampilan data *training* dan data *testing* dalam bentuk diagram batang dapat dilihat pada gambar 8 di bawah ini.



Gambar 8 Hasil Data *Training* dan Data *Testing* Dalam Bentuk Diagram Batang

Dari presentase diatas dapat dilihat bahwa jumlah data *training* sebanyak 1223 data dan data *testing* sebanyak 306 data. *Naive Bayes Classifier* adalah salah satu algoritma klasifikasi yang berbasis pada teori probabilitas Bayes dengan asumsi bahwa setiap fitur atau kata bersifat independen. Dalam konteks analisis sentimen, algoritma ini digunakan untuk memprediksi kategori sentimen positif, negatif, atau netral dari sebuah teks berdasarkan probabilitas kemunculan kata-kata dalam masing-masing kelas. *Naive Bayes* menghitung peluang suatu teks termasuk dalam suatu kelas dengan mengalikan probabilitas kata-kata yang terdapat dalam teks tersebut terhadap masing-masing kelas sentimen. Kelas dengan probabilitas tertinggi kemudian dipilih sebagai hasil klasifikasi.

Implementasi algoritma dalam penelitian ini dilakukan menggunakan *library MultinomialNB* dari *Scikit-learn* di *Google Colab*. Sebelum dimasukkan ke model, data teks diolah terlebih dahulu menggunakan metode TF-IDF (*Term Frequency-Inverse Document Frequency*) untuk mengubah data teks menjadi bentuk numerik yang merepresentasikan bobot pentingnya setiap kata dalam dokumen. Setelah itu, model dilatih menggunakan data latih yang telah dilabeli, kemudian diuji pada data uji untuk menilai kemampuannya dalam mengklasifikasikan sentimen. Berikut merupakan proses melatih model *Naive bayes* yang dimana menggunakan *MultinomialNB* dapat dilihat pada gambar 9 di bawah ini.

```
# membuat dan melatih model multinomial naive bayes
mnb = MultinomialNB() # Added the assignment operator '='
mnb.fit(X_train_vectorized, y_train)
```

Gambar 9 Proses Melatih *Machine Learning Naive Bayes*

Setelah model *Naive Bayes* diterapkan, dilakukan proses evaluasi untuk menilai kinerja model dalam mengklasifikasikan data uji. Evaluasi ini menggunakan beberapa metrik, yaitu akurasi, presisi, recall, dan F1-score. . Evaluasi model ini digunakan untuk mengukur

seberapa baik model dalam menganalisis sentimen terhadap data dari ulasan *google maps*. Berikut merupakan proses evaluasi model dapat dilihat pada gambar 10 di bawah ini.

```
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, classification_report, confusion_matrix

# Evaluasi model
akurasi = accuracy_score(y_test, y_pred)
presisi = precision_score(y_test, y_pred, average='weighted')
recall = recall_score(y_test, y_pred, average='weighted')
f1 = f1_score(y_test, y_pred, average='weighted')

# Cetak hasil evaluasi
print("Akurasi :", akurasi)
print("Presisi :", presisi)
print("Recall :", recall)
print("F1-score :", f1)
print("\nClassification Report:\n", classification_report(y_test, y_pred))

Akurasi : 0.7095709570957096
Presisi : 0.7466069841561789
Recall : 0.7095709570957096
F1-score : 0.6984954532985985
```

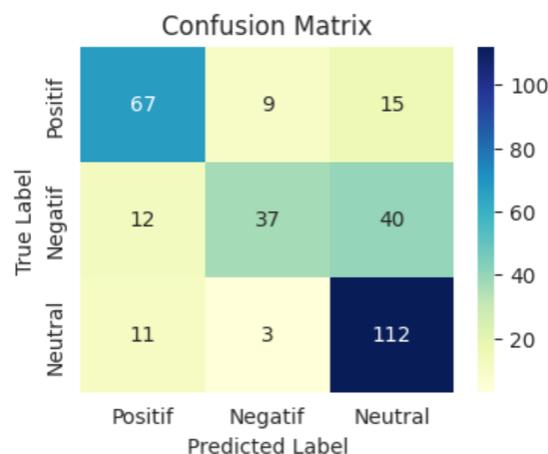
Gambar 10 Proses Evaluasi Model

Kode ini mengevaluasi kinerja model *MultinomialNB* menggunakan metrik akurasi, presisi, recall, dan F1-score. Akurasi mengukur persentase prediksi yang benar, precision menilai ketepatan klasifikasi positif, recall mengukur sejauh mana model mendeteksi sampel positif, dan F1-score menyeimbangkan precision serta *recall*. Berikut merupakan hasil dari evaluasi model yang ditunjukkan pada tabel 11 berikut.

Tabel 4.2 Hasil Evaluasi Model

Accuracy	Precision	Recall	F1-Score
0.709570	0.746606	0.709570	0.698495

Berikut akan ditampilkan *confusion matrix* pada gambar 11 di bawah ini.



Gambar 11 *Confusion Matrix*



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

Hasil evaluasi model klasifikasi sentimen menggunakan *confusion matrix* menunjukkan bahwa model memiliki performa yang cukup baik, terutama dalam mengklasifikasikan sentimen *netral*. Dari total ulasan dengan label *netral*, sebanyak 112 data berhasil diklasifikasikan dengan benar. Untuk sentimen *positif*, sebanyak 67 data diklasifikasikan secara tepat, meskipun masih terdapat beberapa kesalahan klasifikasi ke kelas *negatif* dan *netral*. Sementara itu, model mengalami kesulitan dalam membedakan sentimen *negatif*, terlihat dari 40 data berlabel *negatif* yang justru diklasifikasikan sebagai *netral*. Kesalahan klasifikasi antar kelas ini menunjukkan bahwa meskipun model secara umum mampu mengenali pola sentimen, khususnya pada sentimen *positif* dan *netral*, namun masih terdapat ruang perbaikan dalam membedakan antara sentimen *negatif* dan *netral* yang memiliki kecenderungan makna yang lebih dekat.

Visualisasi *Word Cloud* digunakan dalam penelitian ini untuk menampilkan kata-kata yang paling sering muncul dalam data ulasan. *Word Cloud* menyajikan kata-kata dalam berbagai ukuran, di mana ukuran kata mencerminkan tingkat frekuensinya dalam kumpulan data. Kata yang lebih sering muncul akan ditampilkan dengan ukuran lebih besar. Visualisasi ini membantu memperkuat pemahaman terhadap pola bahasa yang digunakan oleh pengguna dalam ulasan mereka. Berikut merupakan tampilan *word cloud* dapat di lihat pada gambar 12.



Gambar 12 Hasil *Word Cloud*

#### 4. Kesimpulan

Berdasarkan penelitian yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut.

Penerapan algoritma *Naive Bayes* berhasil mengklasifikasikan ulasan pengunjung Sirkuit Mandalika dari *Google Maps* ke dalam tiga kategori sentimen, yaitu *positif*, *negatif*, dan *netral*. Proses klasifikasi dilakukan melalui tahapan preprocessing, pelabelan berbasis *lexicon* (kamus positif dan negatif), serta pembobotan menggunakan metode TF-IDF.

Hasil evaluasi model klasifikasi menggunakan algoritma *Naive Bayes* menunjukkan bahwa performa model berada pada kategori cukup baik. Model berhasil mencapai akurasi sebesar 70,96%, dengan nilai presisi 74,66%, recall 70,96%, dan F1-score 69,84%. Capaian ini mencerminkan bahwa model mampu melakukan klasifikasi sentimen secara seimbang antara ketepatan dan kemampuan mengenali data yang relevan. Meskipun demikian, masih ditemukan kekeliruan klasifikasi, khususnya pada sentimen negatif yang sering kali diklasifikasikan sebagai netral. Oleh karena itu, diperlukan pengembangan lebih lanjut, baik dalam tahap prapemrosesan maupun eksplorasi model lain yang lebih kompleks, guna meningkatkan kualitas hasil klasifikasi.

Visualisasi data menggunakan *Word Cloud* mampu memperlihatkan kata-kata yang paling sering digunakan oleh pengunjung, sehingga dapat memperkuat pemahaman terhadap pola sentimen yang muncul dalam ulasan.



**REFERENSI**

- Darwis, D., Siskawati, N., dkk. 2021. "Penerapan Algoritma Naive Bayes Untuk Analisis Sentimen Review Data Twitter Bmkg Nasional". *Jurnal Tekno Kompak*, 15(1), 131. <https://doi.org/10.33365/jtk.v15i1.744>.
- Europa Publications 2022. *The Europa Directory of International Organizations 2022. The Europa Directory of International Organizations 2022*. London: Routledge, <https://doi.org/10.4324/9781003292548>.
- Fatikhah, T. Z., Muhtarom, M., dkk. 2024. "ANALISIS SENTIMEN ULASAN MIE GACOAN SOLO VETERAN DI GOOGLE MAPS MENGGUNAKAN", 6(3), 649–658.
- Haq, C. S. Octiva, A. Ayuliana, U. W. Nuryanto, and D. Suryadi, "Algoritma Naive Bayes untuk Mengidentifikasi Hoaks di Media Sosial," *J. Minfo Polgan*, vol. 13, no. 1, pp. 1079–1084, 2024, doi: 10.33395/jmp.v13i1.13937.
- Irvandi, Irawan, B., dkk. 2023. "Naive Bayes Dan Wordcloud Untuk Analisis Sentimen Wisata Halal Pulau Lombok". *INFOTECH journal*, 9(1), 236–242. <https://doi.org/10.31949/infotech.v9i1.5322>.
- Ipmawati, J., Saifulloh, S., dkk. 2024. "Analisis Sentimen Tempat Wisata Berdasarkan Ulasan pada Google Maps Menggunakan Algoritma Support Vector Machine". *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 4(1), 247–256. <https://doi.org/10.57152/malcom.v4i1.1066>.
- Prakoso, C., dan Hermawan, A. 2023. "Perbandingan Model Machine Learning dalam Analisis Sentimen Ulasan Pengunjung Keraton Yogyakarta pada Google Maps". *Kajian Ilmiah Informatika dan Komputer*, 4(3), 1292–1302. <https://doi.org/10.30865/klik.v4i3.1419>.
- Rizkina, N. Q., dan Hasan, F. N. 2023. "Analisis Sentimen Komentar Netizen Terhadap Pembubaran Konser NCT 127 Menggunakan Metode Naive Bayes". *Journal of Information System Research (JOSH)*, 4(4), 1136–1144. <https://doi.org/10.47065/josh.v4i4.3803>.
- Rohmania, D. R., dan Abidin, R. 2024. "Analisis Sentimen Masyarakat Indonesia Terhadap Kebijakan Program Tapera Menggunakan Brand24", 1(2), 120–131.
- Undap, M. G., Rantung, V. P., dkk. (t.t.). Analisis Sentimen Situs Pembajak Artikel Penelitian Menggunakan Metode Lexicon-Based. *JOINTER-JOURNAL OF INFORMATICS ENGINEERING*, 02.

**Lisensi**

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.